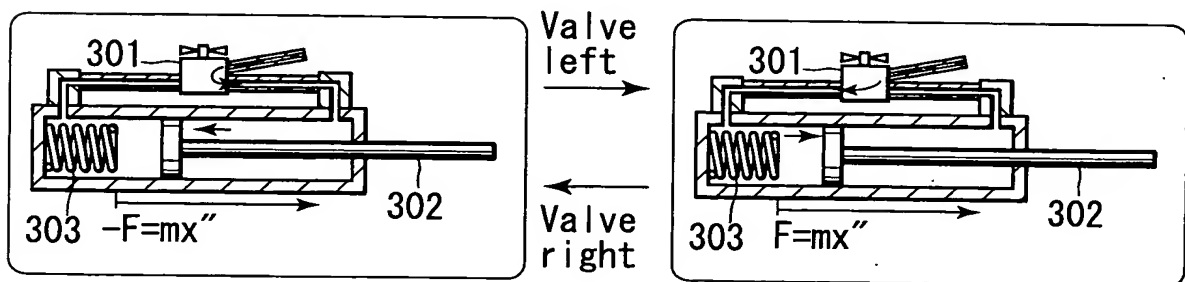
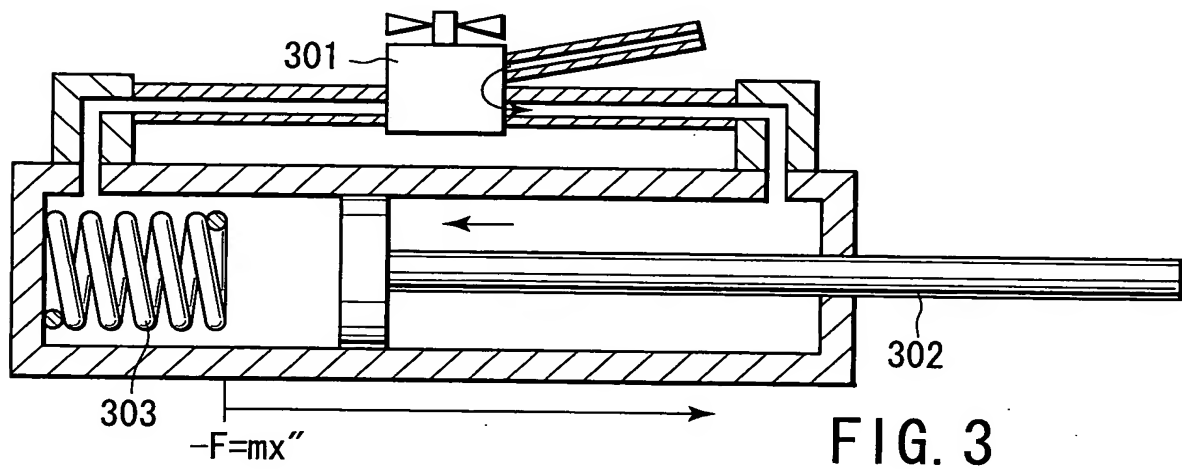
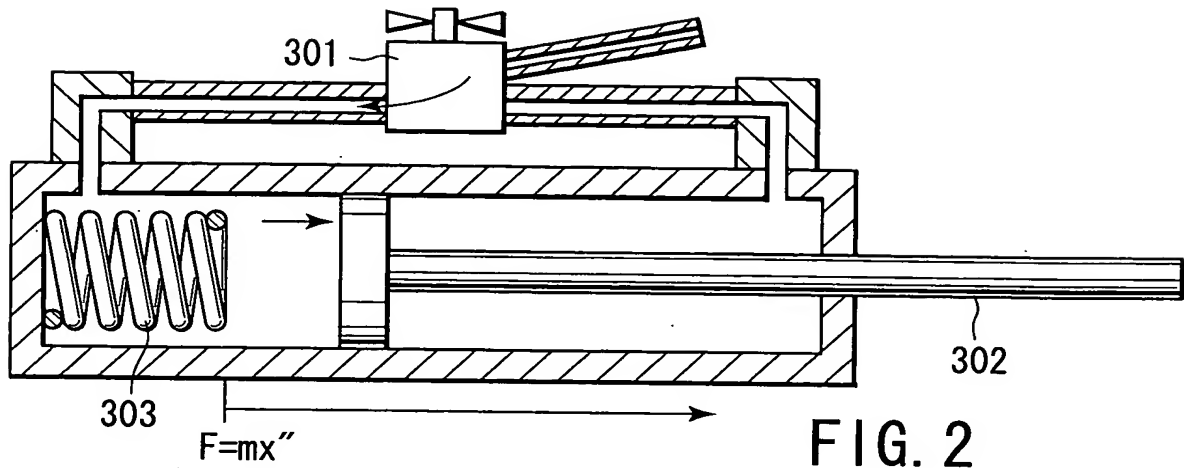


FIG. 1



L1	#define m 1	
L2	#define f 100	
L3	Right, ev1	ev2
L4	wait 50 do Left, ev3	eq1
L5	always if Left then do always F=m * x" watching Right,	
L6	always if Right then do always -F=m * x" watching Left,	
L7	sample(x), ev4	eq2
L8	x=0, x' =0,	
L9	module "C"	
L10	{	
L11	int cPrint (int num)	
L12	{	
L13	FILE* fp=fopen("log.txt", "a");	
L14	fprintf (fp, "%d", num);	
L15	fclose(fp);	
L16	}	
L17	},	
L18	int cPrint (int num),	
L19	wait 100 do E, ev5	
L20	prosess (E) {	
L21	cPrint(x);	
L23	}	

FIG. 5

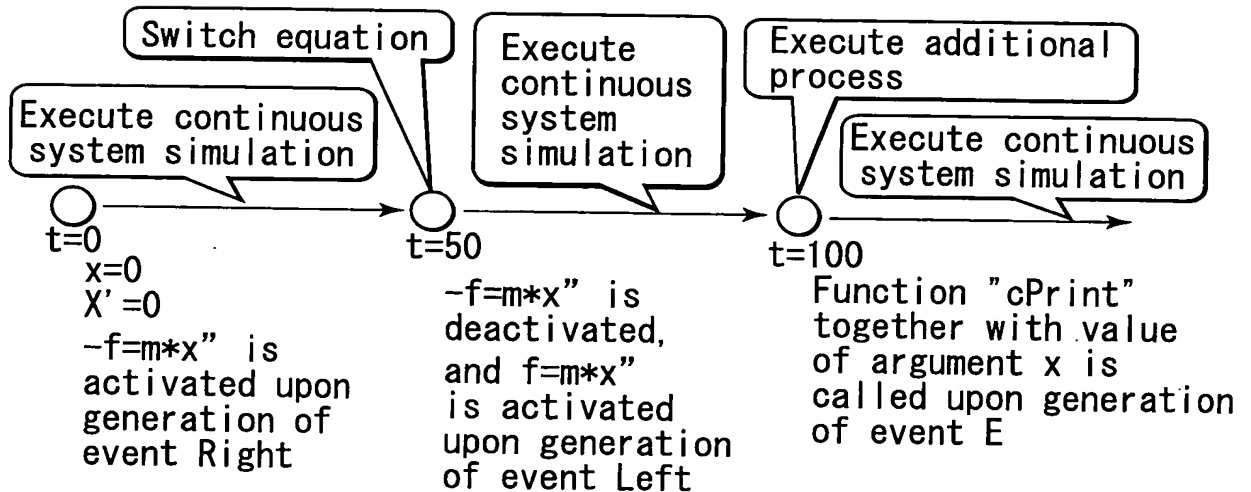
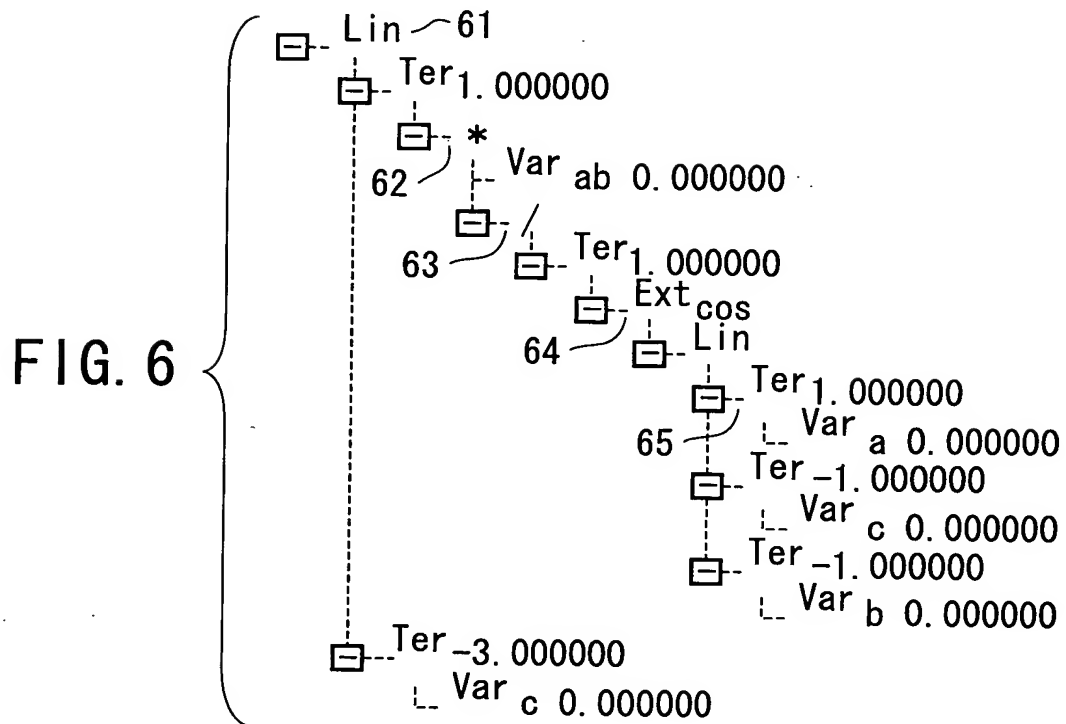


FIG. 8

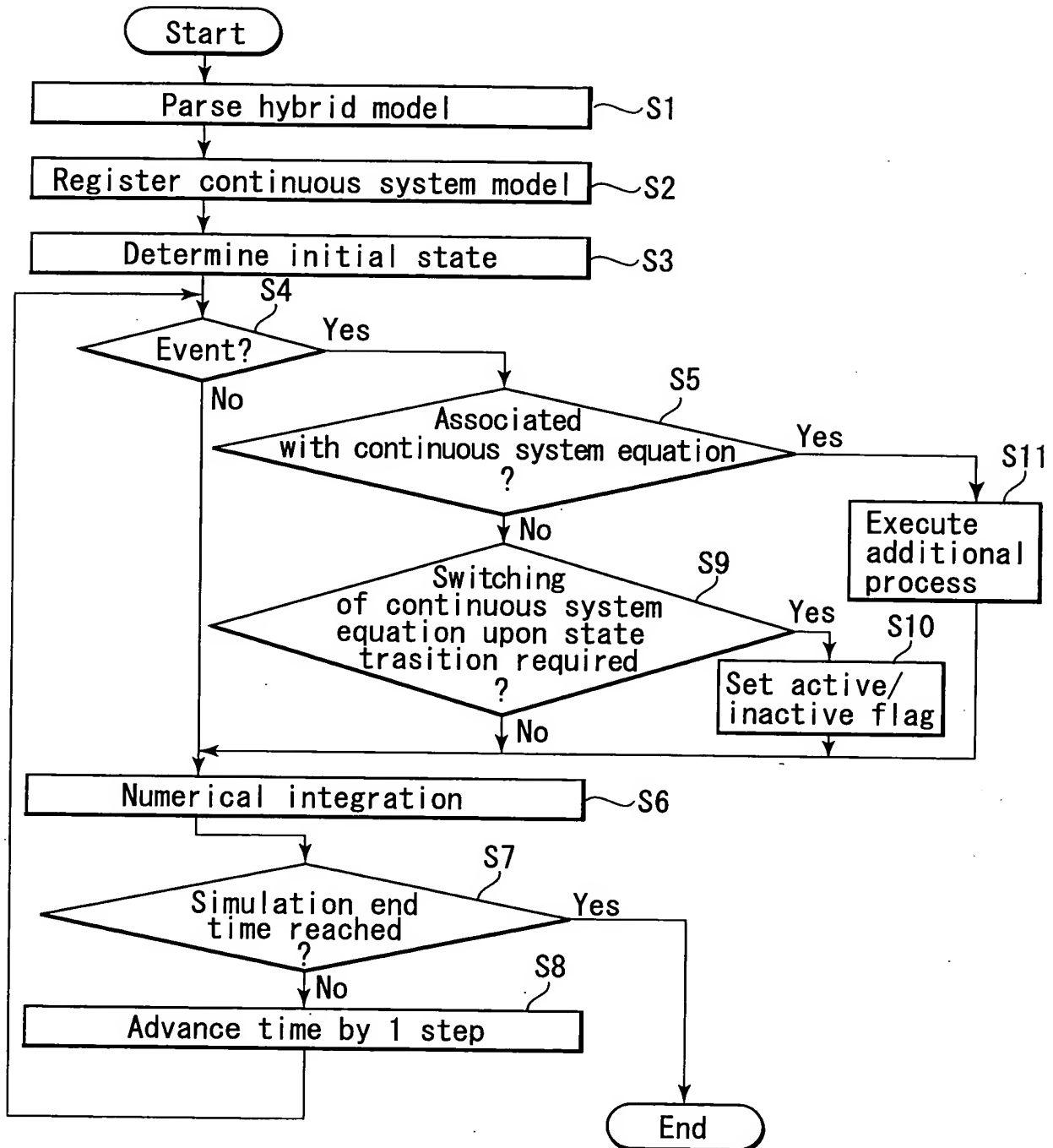
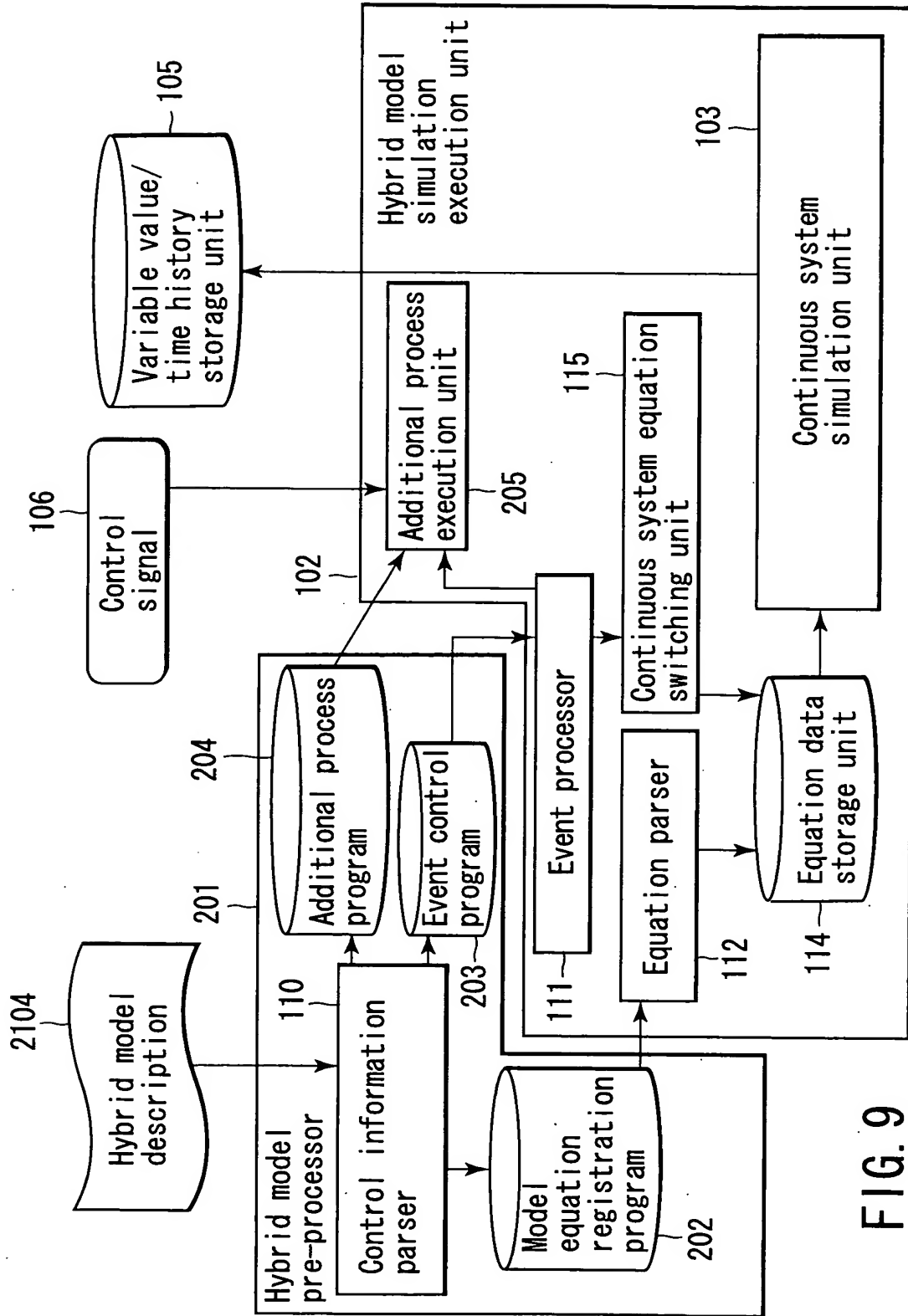


FIG. 7



```
1  #define m 1
2  #define f 100
3  Right,
4  wait 50 do Left,
5  always if Left then do always  $F=m*x$ " watching Right,
6  always if Right then do always  $-F=m*x$ " watching Left,
7  sample(x),
8  x=0, x'=0,
9  module "C"
10 {
11     int setDataToCtrl( int num, int data )
12     {
13         outport(num, data);
14     }
15     int getDataFromCtrl( int num )
16     {
17         return inport(num);
18     }
19 },
20 int setDataToCtrl( int num, int data ),
21 int getDataFromCtrl( int num ),
22 wait 100 do E1, wait 150 do E2,
23 process( E1 ) {
24     setDataToCtrl( 1, x )
25 },
26 process( E2 ) {
27     x=getDataFromCtrl( 1 )
28 }
```

FIG. 10

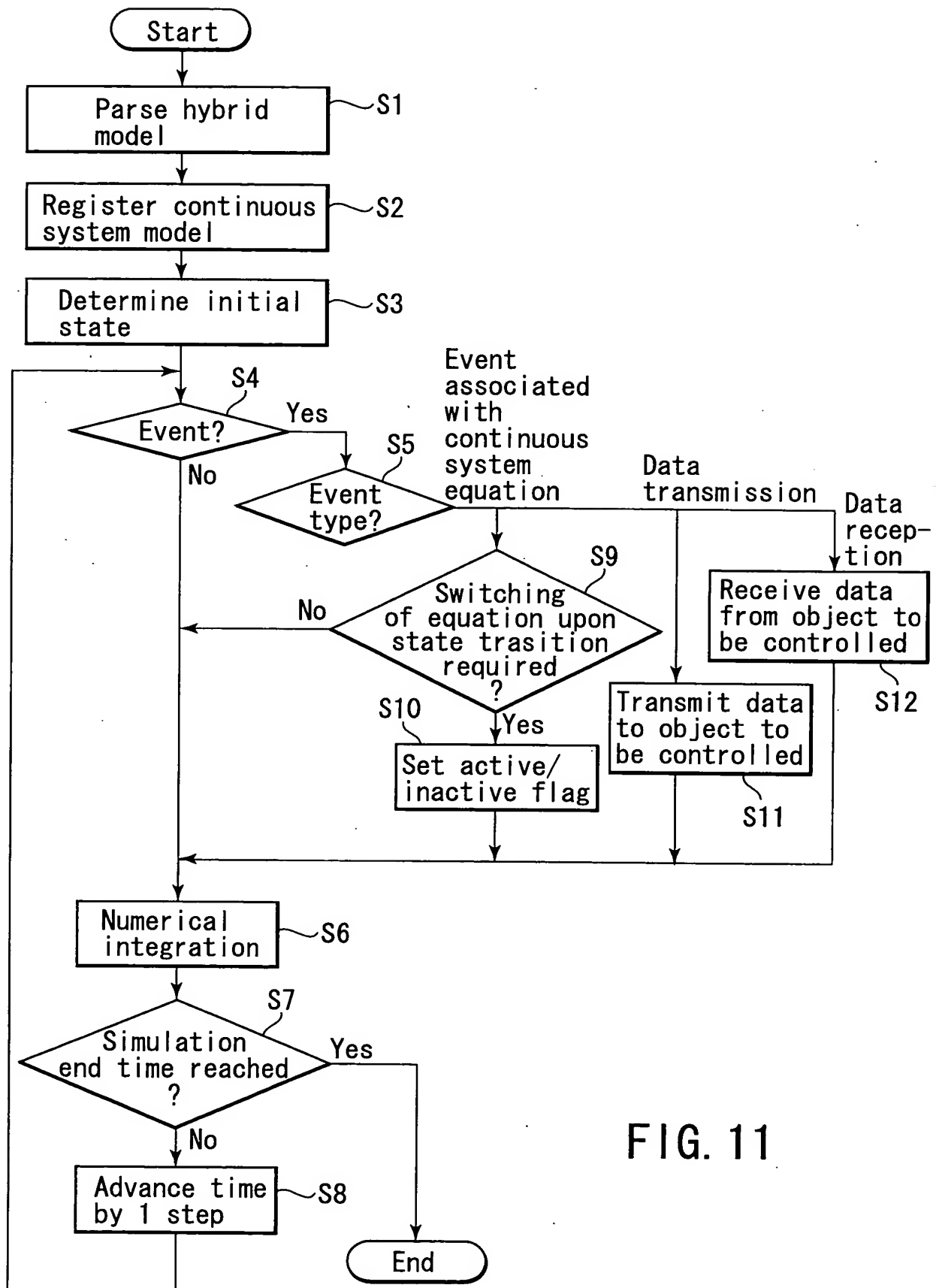


FIG. 11